J. M. Coetzee's aesthetic automatism

Roach, Rebecca Claire

DOI:
10.1353/mfs.2019.0015

Document Version
Peer reviewed version

Citation for published version (Harvard):
Roach, RC 2019, 'J. M. Coetzee's aesthetic automatism', *MFS - Modern Fiction Studies*, vol. 65, no. 2, pp. 308-337. https://doi.org/10.1353/mfs.2019.0015

"J. M. Coetzee's Aesthetic Automatism"

[insert fig 1.]

At some point in 1962 or 1963 the document depicted in Figure One was created. On first glance it is a page of handwritten notes, taken on consulting the abstract of "Joseph Conrad and Ford Madox Ford: A Study in Collaboration" (1960), an unpublished dissertation ("Dissertation Abstracts"). The abstract itself was printed in *Dissertation Abstracts*, a bibliography of American doctoral theses published annually by University Microfilms International; full dissertations could be purchased in photocopy or microfilm form, as the handwritten notes record. Looking more closely, the notes were written on perforated paper, more commonly deployed to record the printed output of a computer and at the top of the page two half-lines of code are just visible. Holograph, microfilm, Xerox, computer: the document in Figure 1 beautifully illustrates the complex ecosystem of inscription and storage technologies existing in the early 1960s; a moment when computers relied on punch-cards, paper print outs and magnetic tape and when digital computer and paper were not yet conceptually opposed.

The above description could be the opening paragraph in what would be a fascinating supplement to Lisa Gitelman's work on the media history of documents. In speaking of this object, my purpose is slightly different. The handwriting, as the title of this article suggests, is that of J. M. Coetzee. The notes were written when Coetzee was researching a thesis on Ford Madox Ford at the British Museum Reading Room in London as part of an MA he was completing at the University of Cape Town. Simultaneously, he was also working for the British arm of the pre-eminent

American computing firm, IBM, as a programmer. While Coetzee's reputation rests on his prize-winning novels, autobiographies and critical writings, less well known is his practical and intellectual engagement with computing over a period of more than fifty years and with his role on one of Britain's most advanced computing projects of the 1960s. Readers may be familiar with a version of this history from Coetzee's second "fictional autobiography" *Youth* (2002); however, there are some notable deviations between this fiction and what we might term the "archival" record. Here I am interested in what light this paper, punch card, and magnetic tape trail might shed on Coetzee's development as a writer and a critic.

Recounting the specific manner of Coetzee's engagement, this article argues that these years were to have a foundational influence upon his later writing. Critics have often noted Coetzee's background in mathematics; taking their cue from the writer, who has himself argued for the parallels between mathematics and poetry as "two rarefied forms of symbolic activity", they frame the relationship as one of conceptual influence (Review of *Strange Attractors* 944). However, as Coetzee's biographer mused, "One wonders ... how congenial the abstract ratiocination of mathematics can be for the verbal artist who draws on the concrete world for his raw material and means of representation, and who has to think with his senses" (Kannemeyer 88). Taking a more historical and material perspective informed by work in critical code studies, software studies, platform studies and media archaeology, which as Lori Emerson has argued, demonstrate the "continued relevance of thinking through intentionality alongside materiality" (51), I deliberately focus on the crucial early years of Coetzee's engagement with applied mathematics

and the material affordances of computing.[1] Ironically, given what Lev Manovich has seen as the neglect of "formalism" in media studies, I am deploying this perspective to argue that this engagement would result in Coetzee's development of a platform of "aesthetic automatism". My term draws on what Coetzee would describe, with deep ambivalence, as the Prague school notion of "automatization", "the process by which repeatedly used speech forms wear a neural rut". Developing out of, and yet often in opposition to, the idea that "Automatized speech is speech that speaks its speaker", Coetzee's platform of aesthetic automatism, as I conceive it, involved a conception of literary form and language that sought to recast modernism's aesthetic autonomy within a mid century context in which computing formalisations were automating many processes, including those of reading and writing ("The First Sentence" 93).[2]

The development of this platform indeed forms the basis of *Youth*, the fictional narrative of an "ignorant provincial" and aspirant poet, struggling miserably to find his literary voice among a cacophony of inherited aestheticist and modernist strictures (51). It is also a work seeking to resolve such demands for modernist impersonality, verbal "compression" and "fine discriminations" within the context of

[1] For an excellent overview of such work and its implications for scholars of book history, see Kirschenbaum and Werner. Coetzee's computing experience is notable for occurring at a moment before the distinctions on which platform, software and critical code studies rely today were clearly articulated. My use of the term "material" is inflected by the work of scholars such as Lisa Gitelman, Matthew Kirschenbaum and N. Katherine Hayles working at the intersection of media archaeology, media theory and literary studies in an attempt to counter the trend by which "literary criticism has for much too long tended to regard the literary work as an immaterial verbal construct" (Gitelman "Materiality", 8). I use the term "formalisation" to refer to a process of de-materialisation, or the move to abstract away from materiality; I do, however, consider form to be entangled with materiality and am guided by Caroline Levine's understanding of forms as structuring patterns, as constraining, as doing political work in historical contexts, and as having particular affordances (Levine 14–23).

[2] In arguing for Coetzee's platform as one that sought to recast modernist autonomy, I don't wish to downplay the complexity of his own relationship to literary modernism (in its various incarnations), a topic that has itself proven particularly fertile for scholars including Zimbler, James and Attridge.

automation; as the narrator ponders, "might it not be argued that the invention of computers has changed the nature of art, by making the author and the condition of the author's heart irrelevant?" (61, 135, 161) Although written as a fictional account almost forty years after his time working in the industry, *Youth* suggests that readers would do well to consider the idea that Coetzee's "famed impersonality", which as David Attwell notes, is "not an a priori quality inherent in a work of art, nor is it simply a function of the aesthetic. It is an *achievement*", has its roots as much in his work with computer programming as it does in his engagement with high modernism (33).

Certainly modernism's own engagement with new media has been widely discussed in recent scholarship. Jessica Pressman, in particular has argued persuasively that close reading was itself the product of modernism's "mediatized moment" and that Marshall McLuhan's media theory, which was itself to leave a strong trace on Coetzee, was itself an adaptation of the "New Critical method of focusing on form to think critically about forms of media" (11, 25). More particularly, mid century projects such as Printed English and Machine Translation (with its aspiration for a Universal Language) have led scholars including Lydia H. Lui and Heather A. Love to identify an imbrication of modernist and cybernetic ideas, notably around a model of information first articulated by Claude Shannon in his 1948 article "A Mathematical Theory of Communication". In Coetzee's early computational experiments we find an enthusiastic engagement with these formalist projects, a key example of what Hugh Kenner (whom Coetzee met in 1968) would call "art in a closed field", or the relentless drive toward abstraction, via permutation, exhibited in writers such as Beckett and Joyce. Indeed, in 1977 Coetzee would publish an article on the "poetics of failure" he identifies in works such as Beckett's *The Unnamable*,

Nabokov's *Pale Fire,* and Achterberg's "Ballade van de gasfitter", or "program for constructing artifacts out of an endlessly progressive etiolated self-consciousness lost in the labyrinth of language and endlessly failing to erect itself into autonomy" ("Achterberg" 293).

Yet we also find in Coetzee's work a clear articulation of the limits of such projects; limits born not of modernism's "exhaustion", but rather of the automation of language that computational media, via a cybernetic theory of language that is dematerialised and probabilistic, frequently embody and promulgate (John Barth, cited in McHale 28). As we shall see, Coetzee might characterise his art as "Spare prose and a spare, thrifty world", but it is highly constructed, built in part via a deeply material, non-cybernetic understanding of language born of his own engagement with computing (*Doubling the Point* 20).

FORTRAN thinking: Coetzee the Programmer

Coetzee's employment in the computer industry was focused over a four-year period in the early 1960s, when he was based in London. Before immigrating to the United Kingdom, he had graduated from the University of Cape Town (UCT) with honours in both English and Mathematics. His undergraduate experiences were formative for his later interests, as Attwell, Kannemeyer and others have demonstrated. On the literary side he studied stylistics, philology, literature across periods and more contemporary modernist writers and their criticism, as well as non-credit classes in imaginative writing with the poet Guy Howarth (Kannemeyer 89-96). For his mathematics degree, he would study under Douglas Sears and Stanley Skewes, the latter renowned for his discover of the Skewes number and associate of Alan Turing before the war. Coetzee's courses in Pure Mathematics I, II and III, Applied

Mathematics, Mathematical Statistics III and Mathematical Honours would also give him a firm grounding in the ideas that formed the basis of computer work (Kannemeyer 87-89). However, it was in the UK that he first obtained a position as a programmer, in the London office of American computing giant International Business Machines (IBM) as an applications programmer.

This was 1962. The industry was in transition as computer use expanded from origins in military research and mathematical applications, to much wider deployment in business, academic research, and beyond. The social, economic and cultural implications of computation were being hotly contested in the public sphere. At the same time the pace of technological innovation was rapid: it was only eleven years since the delivery of the UNIVAC 1 and the advent of commercial computing (Ceruzzi). In hardware the move had only just been made from vacuum tube to transistors. As the market for computation grew, so did the realisation that programming was a pinch point within the industry: the new markets required expensive specialised software and there was a dearth of good programmers to write it. Part of the difficulty, as Nathan Ensmenger has noted, is that defining and then identifying a good programmer was proving more difficult than anticipated: far from being mere glorified clerical work, good programming was a "black art" (40).[3] In their search for programmers companies like IBM hired people from a diverse range of backgrounds, in particular, poets, chess players and mathematicians. As a chess player, mathematics graduate and aspiring poet, Coetzee was an attractive candidate.

When Coetzee joined IBM the company was the global leader in the booming electronic computer industry, having expanded from being a purveyor of tabulators in the 1950s. In 1960 IBM had launched what would become the classic mainframe

[3] Ensmenger takes the term "black art" from FORTRAN inventor John Backus.

computer: the 7090. Despite being as big as a room and priced at almost $3 million, the computer was a commercial success thanks to its speed, architecture and the flexibility of tasks to which it could be put. The 7094, for example, had a (for the time) huge 36-bit word length, which made it suitable for scientific calculations and a large core memory. Often run in conjunction with the smaller 1401 model, Coetzee spent most of his time working on the latter. Announced in 1959 and designed for the business world, the 1401's popularity demonstrated to the industry the existence of a sizeable commercial market for computers. Within IBM, for reasons of efficiency (time on the 7090 was expensive), programmes would often be developed on the 1401 using punched cards and, once perfected, only then be transferred on to tape and run on the 7090.

Working on these machines, Coetzee would likely have conceived of computing as a multi-media activity with very clear constraints. Far from the immediacy of today's user experience, Coetzee's computer interfacing at IBM involved interacting with a host of different objects: switches and drum storage units, console keyboard, punch cards, magnetic tape and continuous form printer paper. Memory was limited, computation entailed the correct sequential inputting of notched cards for batch processing and programming was based around the linear and temporally discrete structure of such processing: only one programme could be run at a time and it couldn't be altered half way; when it finished the human operator had to reset the machine for the next job. Extremely advanced technology for the time, the computer's physical limitations were nevertheless obvious to those working with them: despite the multiplication of physical objects this was an environment of scarcity. This culture resonates with Coetzee's own conception of the writing craft. His fiction regularly conceives of its medium in terms both minimalist and material:

while his characters desire "a life unmediated by words", Coetzee's austere prose renders these words objects "to burnish and fondle" in *Dusklands* (1974) or stones to be itemised and manipulated into messages in *In the Heart of the Country* (1976) (*In the Heart of the Country* 147; *Dusklands* 43).

Meanwhile the writing of programmes would also, I contend, have ramifications for Coetzee's understanding of literary language and form. Programming was in effect a two-stage process: firstly the design of the programme entailed the establishment of an objective – a calculation – and the logical process needed to arrive at that objective. All assumptions needed to made explicit (unlike humans computers "will embody no *inexplicit* preconceptions" [Needham 429]) and organised in an efficiently written series of logical propositions. This stage could be inscribed graphically in a series of flowcharts, the advantage of such "diagrammatic methods" being, as Coetzee would later acknowledge in his doctoral thesis, that they "have the power to condense information which language seldom equals" ("The English Fiction" 148-49). Alternatively, these stages could be written in "pseudocode", an informal and often idiosyncratic notation system that uses the structural conventions of programming languages but is not executable and is designed to read by humans, not computers. Only after this step could the programme, the embodiment of the intention, be coded into a machine-readable language for the computer to execute.

When Coetzee was at IBM, programmes were usually written in FORTRAN. The first high-level language, FORmula TRANslation was developed in 1957. Computer instructions had previously been written in so-called "machine code", which consisted of strings of binary or hexadecimal numbers. These numbers are executable by the computer, but difficult for the human coder to write and read.

Machine code is also specific to the individual computer. With the advent of compilers, which automated code conversion, programmers could write the code in a more human-friendly language. This made coding both more efficient and more accessible to scientists and mathematicians, reliant as it was on mathematical formulae. FORTRAN, like most formal languages, deviated from natural language in notable ways: it was designed to avoid ambiguity and redundancy, it also promoted literalness. FORTRAN was also significant for its emphasis on mathematical and logical propositions. In using FORTRAN, Coetzee was working at one level of abstraction from the computer's machine code, in a language that utilised arithmetical statements, efficiency statements, and sub programs (*Reference Manual* 5). While it could be intellectually stimulating, both stages of programming involved rule-bound composition that was highly reliant on the either-or logic of Boolean algebra, that drew on a limited range of statements, and which was oriented towards executable function. Such experience likely prompted Coetzee to consider the operations of natural language and the precise functions of literary language.

Overall Coetzee seems to have found working at IBM to be intellectually restrictive. This is in direct opposition to much of the cultural hype around a company whose mantra was "THINK". While adopted in 1914, as *Time* triumphantly declared in 1955, IBM "hopes to mechanize hundreds of processes.... Thus liberated from grinding routine, man can put his own brain to work on problems requiring a function beyond the capabilities of the machine: creative thought" (quoted in Maney et al. 16). As an employee, rather than a customer, Coetzee was not "liberated from" but in service to the "grinding routine" of the computer. In letters Coetzee described the experience as "sterile" (Letter from Coetzee to Stanley Skewes). In *Youth* too, the narrator is damning of the entire enterprise: "The more he has to do with computing,

the more it seems to him like chess: a tight little world defined by made-up rules" (149).

No surprise then, that in 1963 Coetzee resigned from IBM. Returning temporarily to South Africa, he was still intent on returning and pursued various employment opportunities back in the UK. While exploring the option of teaching, he also wrote to IBM's competitors, including International Computers & Tabulators (ICT), Burroughs Machines Ltd, English Electric-Leo Computers Ltd, and the University of London, suggesting Coetzee disliked IBM rather than the industry as a whole. In January 1964 he was offered the job of programmer on the Atlas project by ICT, which he accepted.

Babeling in a Closed World: the Atlas Project

Atlas was Britain's first supercomputer. A joint development between the University of Manchester and Ferranti, an electronics company with numerous defence contracts, the computer was designed between 1956 and 1962 under the lead of Tom Kilburn. The computer was not only fast, but introduced various innovations, including virtual memory and a multitasking operating system, called the "Supervisor". Although technically impressive, (arguably the fastest in the world at its commissioning in 1962) only three Atlas computers were delivered, mainly due to their prohibitively high cost. Ferranti was interested in producing a simpler and more commercially viable model; in 1962 the company supplied the University of Cambridge's Mathematics Laboratory with Atlas hardware at discounted rates in return for research support. Called Titan, this machine was the prototype for the Atlas 2 supercomputer. It was on the Titan/Atlas 2 project that Coetzee was employed as a programmer (in 1963 Ferranti's mainframe computer interests had been absorbed by ICT). A far cry

from the routine business processing that occupied him at IBM, the work that Coetzee was doing as part of the Titan/Atlas 2 project was at the cutting edge of computing. He worked closely with a team comprised of computer scientists and mathematicians from both ICT and Cambridge University, a veritable who's who of Britain's post-war computing luminaries, including Roger Needham, David Wheeler, David Hartley, David Barron, Peter Swinnerton-Dyer and Barry Landy.

Coetzee was part of a team working on the Titan's Supervisor. Unlike previous computers, the Atlas 1 and Titan deployed multiprogramming (i.e. several programmes could seemingly be run concurrently). The Atlas Supervisor was the first computer operating system; if previously a computer was "effectively [an] empty machine", the Supervisor offered "a radical solution to this situation, integrating the treatment of ... input/output handling" (Pyle 7). The programming entailed in developing this operating system was far from minor: 35,000 machine instructions according to computer historian Simon Lavington ("The Atlas story" 21). When it came to the Titan/Atlas 2, an entirely new operating system was needed: the Atlas Supervisor had exploited the computer's innovative one-level store, (virtual memory and paging), however, to save on costs this feature was removed in its successor. The Titan/Atlas 2 Supervisor had to be designed from scratch.

Coetzee's work entailed conceptualising the design and writing the code that would instruct the computer about which jobs to implement and in what order. In *Youth* he satirises such work on this machine with "self-consciousness of a kind": "At regular intervals – every ten seconds, or even every second – it interrogates itself, asking itself what tasks it is performing and whether it is performing them with optimal efficiency." With each "swing of the magnetic tape" the operating system will question itself, moving to the rhythm of the pendulum, the arms race and the march

towards automation (143). Nevertheless, whatever the goal of job control the actual experience of programming on the Titan/Atlas 2 Project was far more provisional, messy, and human.

One of the most notable features of working on this project was its Babel-like multilingualism and coding inventiveness. This was a moment in which programming languages were proliferating thanks to the development of compilers but before these languages were defined by formal standards (the first was FORTRAN 66 in 1966). The collaborative but federated nature of the Atlas project meant that it was particularly notable for its programming language multiplicity and creativity. The original Atlas machines had variously created and deployed Mercury Autocode, Atlas Autocode for the Manchester Atlas 1, a compiler-compiler for the University of London Atlas and a FORTRAN dialect and compiler for the Harwell Atlas. Meanwhile the Titan utilised its own autocode developed by David Hartley while the new language it was to run on, CPL, was in development. Coetzee, thanks to his prior work with FORTRAN, had expertise in programming with the most advanced formal language, while also working with machine code, with languages under development and with programmes written for the other Atlas machines in various other dialects and languages.

Working across these languages, Coetzee would likely have been well aware of the degree to which the specific style and structure of these early formal languages affected not only the code that could be written, but also the computational functions that could be performed. As Donald E. Knuth and Luis Trabb Pardo note in their 1976 article on the "pre-Babel" days of programming: "After learning a high-level language, a person often tends to think mostly of improvements he or she would like to see (since all languages can be improved), and it is very easy to underestimate the

difficulty of creating that language in the first place" (2). Coetzee was uniquely positioned to experience the process of language formalisation. Moreover, in moving between abstracted languages and machine code, Coetzee was likely confronted on daily basis with questions around the material implications of working within formalisations and at varying degrees of abstraction. These practical questions would lead directly to his own doctoral work in stylostatistics and to a life-long interest in the material origins and import of linguistic structures.

One final feature of Coetzee's employment at ICT is vital to note. In response to innovative work being done contemporaneously at MIT, the Titan team began developing one of the earliest time-sharing systems for the Supervisor. Time-sharing built on multiprogramming, enabling multiple users to interact concurrently (so it seemed) with the same computer. The work on time-sharing, however, brought with it a shift in Coetzee's responsibilities – and a raise. In addition to Titan, two Atlas 2 computers were being built, one of which was destined for the Atomic Weapon Research Establishment at Aldermaston. For security reasons, Aldermaston was not keen to have more than one job stored in the computer's memory at any one time; time-sharing was "a complete No No" (Landy, "Atlas 2" 6). The Aldermaston Supervisor would thus need to deviate from the operating system being designed for Titan. In the spring of 1965 the Cambridge and ICT programmers therefore split into separate teams in order to focus on the two different iterations of the Supervisor. For the remainder of his time at ICT, Coetzee's programming activities were focused on the Aldermaston Atlas 2.

The shift in his work had, I contend, long-term consequences for Coetzee's thinking about computing, underlining as it did the connections between his own work on multiprogramming and its wider social, ethical and political implications.

Aldermaston was a high security environment, where much of the government's top-secret research into nuclear armaments was being conducted. Much of this research remains classified and relied on impressive computation facilities. Aldermaston had received one of the original Ferranti Mark 1 computers and by the time Coetzee was working on the project, in addition to the Atlas 2, Aldermaston also had an IBM Stretch supercomputer, along with (in an era before ARPANET, a precursor of the Internet) remote electronic access between the computers and other sites (Lavington, "Ferranti" 4, 8). These were innovative and expensive technologies, and, for the companies providing them, high stakes contracts. Teams from Ferranti/ICT, IBM, Aldermaston and the Atomic Energy Research Establishment collaborated extensively around both hardware and programming to ensure these machines worked effectively – notably with the development of several FORTRAN compilers – and to ensure that Britain remained at the forefront of atomic weapons research. Aldermaston represented the embodiment of what Eisenhower called in 1961 the "military-industrial complex".[4]

Notably, in the late 1950s and 1960s the so-called Aldermaston March was the centrepiece in the Campaign for Nuclear Disarmament's annual calendar. The march between London and Aldermaston in Berkshire was used to raise awareness and protest against global nuclear armament, drawing upwards of sixty thousands of people at its height. Far from a secret government facility, Aldermaston was a politically charged, visible symbol of the military-industrial complex and Cold War nuclear armament. In working on the Atlas 2 at Aldermaston, Coetzee was employed

[4] While *Youth*'s narrator helps a Mr Pompret with "wind-tunnel data" for the development of the TSR-2 bomber, I have found no evidence of this in Coetzee's own employment record, nor in publically available records. IBM did not respond to my request for information. Damien Burke, author of *TSR2: Britain's Lost Bomber* (Ramsbury: Crowood, 2010) found no mention of IBM in his research (although he notes that IBM may have been a supplier to a supplier) (Burke, 2017).

on one of the more controversial technical projects of his era and seeing first hand the ways in which computing was imbricated with – and facilitating – the politics of what Paul N. Edwards memorably described as a "closed world" (Edwards). When he came to write "The Vietnam Project", Coetzee's depiction of the war as the culmination of control by "the voice of the master of statistics" and a "Supervisor" (named Coetzee) schooled in "game theory", was directly drawing on his own knowledge of the computational processes lying behind the military-industrial complex and the closed world they produced (*Dusklands* 14, 32).

Coetzee's role on the Atlas project has been largely ignored by literary scholars and historians of computing alike. Lack of interest by the latter group can be explained partly due to Coetzee's decision to leave the Atlas project before its completion.[5] As the most exciting computer research project of the time in Britain, his move across the Atlantic to pursue a PhD in English is often interpreted as a severing of connections with the world of computing and evidence of a general disinterest in the discipline. This is despite the fact that his subsequent scholarly and creative work testifies to continued engagement with concerns developed while employed in the computing industry. Literary scholars have also downplayed the influence of this period on his intellectual development. Specifically, the practicalities of working in the computer industry in the mid 1960s tend to get overlooked. Where his work is discussed, it tends to be within a wider argument about Coetzee's engagement with mathematics, specifically, pure mathematics.[6] While this subject is clearly important to Coetzee, in ignoring the realities of the computer industry, with its applied use of mathematics, linguistic formalisation and creativity, ties with business and defence,

[5] Notably there is no mention of Coetzee's role on the Atlas project in any of the Atlas Symposium reminiscences or published histories.
[6] The most extensive treatment is Johnston's "'Presences of the Infinite'."

and its multi-media environment, scholars miss the specificity of Coetzee's experiences and their implications for his writing.

Dark Art and Computers in the Reading Room

At the end of August 1965 Coetzee sailed for the United States to start a PhD in English at the University of Texas at Austin; but this was not the radical change in direction that it might first appear. While it is generally known that Coetzee was writing an MA thesis on the writings of Ford Madox Ford during the period in which he was working as a programmer, and for some scholars implying a certain pragmatism in Coetzee's choice of career (paying the artist's food bill), his reading was as much guided by his day job as his dissertation. In fact, Coetzee's intellectual interests over the course of his doctoral study and as an early career researcher in the late 1960s and 1970s grew, at least in part, out of the work he was doing at IBM and ICT and the seeming strain that work placed on modernist and New Critical conceptions of authorial impersonality, close reading practices and form.

On first glance, Coetzee's dissertation topic might suggest that his reading was little concerned with computing, focusing as it did on Ford's literary impressionism. Quoting Ford's own declared preference for "Impressions rather than Statistics", Coetzee puts forward the Poundian argument that "selection" is the key to creativity. However, his conclusion that *The Good Soldier* was "probably the finest example of literary pure mathematics in English" hints at Coetzee's struggle to resolve much broader question, namely whether "impressions or statistics might provide the most authentic 'means of approaching to the heart of things'" which, as Peter Johnston has argued, "coloured almost every aspect of his literary, linguistic, and ultimately ethical

enquiry" in the 1960s and 1970s (Coetzee, "The Word of Ford" x; Johnston, "Presences of the Infinite" 68).

Much of this interest is evidenced in the notes that Coetzee took when reading for the project, many of them written on IBM's green and white printer paper. Along with quotations concerning literary impressionism – Ford's conclusion that "Mr James has carried the power of selection so far that he can create an impression with nothing at all" (Coetzee, "Ford Madox Ford") – Coetzee's reading in the British Museum and across the 1960s included books engaged with questions around the representational and conceptual implications of a shifting media ecology. He consulted works such as Hugh Kenner's *Flaubert, Joyce and Beckett* (1962), Marshall McLuhan's *The Gutenberg Galaxy* (1962), Sigfried Giedion's *Mechanization Takes Control* (1948), G. T. Guilbaud's *What is Cybernetics?* (1960) and Walter J. Ong's *Ramus, Method and the Decay of Dialogue* (1958). He followed work in the emerging field of cognitive science and his notes testify to his personal interest in forms of information representation: recording quotations from Christopher Alexander's *Notes on the Synthesis of Form* (1964) for example, which used a mathematical framework to discuss design process, Coetzee made notes in block diagrams, rather than alphabetic script ("Christopher Alexander"). A habit that he would retain in the years to come, Coetzee's overriding interest was in the relationship between cognition and formal systems of representation, their media instantiation and the implications of this for the literary and critical craft. His thesis might have been concerned with Ford's impressionism, but his context for this discussion was much broader, stemming very explicitly from the tensions his work in programming exposed.

Coetzee's intellectual engagement with the implications of formalisation extended beyond the reading room. During his time in the UK a small but growing

community of artists and scholars had begun to explore the material and conceptual capacities of computers to generate and analyse artistic output. Coetzee would follow this community closely and used his employer's equipment to experiment with the artistic possibilities that computer technology offered to a young writer striving to find a literary platform and voice.

The first of his experiments was published in the March-April 1963 edition of a UCT student magazine, *The Lion & the Impala*. The "Computer Poem", which Coetzee explicitly states was written on a 1401 computer, was the result of a "primitive" programme and his own editing. The published piece comprises the un-edited and edited poems, along with a glib commentary, which is in part exegesis of the process of creation. As Coetzee explains, the programme utilised a pre-set vocabulary (in this case 800 words from Roget's *Thesaurus*) and structure to generate a list of all possible variants of word choice within these confines – in this case 2,100 poems printed at a rate of 75 per minute. Coetzee then "wade[s]" though this "garbage", selecting and editing the output.

Poem (ex computer)		Poem (edited)
	Dawn Birds Stream	Dawn, birds, a stream, a calm morning.
	Calm-Mornings	You stand among the trees alone and
You)	Stand-Among	/tense.
	Forest	You have cried.
	Alone Tense	You spend the nights away from me,
You)	Cry	Terrified, rapt,
You)	Spend-The-Nights	Among owls and black men,
I)	Away-From	Hoping for violence.

Terrified Rapt

Owls Blackmen

You) Hope Violence

Retaining the "key words" so as not to "dilute" the "'originality'" of the poem, Coetzee privileges semantics (and pronouns) over syntax. He happily shifts "You) Cry" from simple present to present perfect, thus interrupting the potential triple repetition, while retaining the majority of word choices. Overall, the grammatical shifts and the poem, with its theme of "personal estrangement", are less interesting than the cynicism expressed in the commentary as to the role of the editor, the programme, the computer and the status of the work itself. The programme in its current iteration is dismissible for Coetzee, requiring an editor with "elementary critical abilities"; however, he does foresee a time in which a sophisticated "hack" could run the programme, producing a decent poem without relying on editorial intervention ("Computer Poem" 12-13).

This published poem and its exegesis was not the only literary experiment that Coetzee undertook while at IBM. His archives include several print outs from this period; notably in May 1963 he was writing a poetry "Line Generator" for an IBM computer, most likely the 1401. Similar to the previous programme, this line generator utilised a library of 1000 words, with the vocabulary assigned a grammatical class (pronoun, infinitive verb, preposition, noun, modifier):

WE ROAM / WITH SKY / COLD

WE WANDER / ABOUT CITIES / WAITNG

YOU WAIT / BETWEN LEAVES / BURNNG [7]

The 508 lines testify to the material constraints within which Coetzee was operating: the 1401 utilised 8-bit bytes, meaning that each byte could contain 6 characters (the other 2 being reserved for word mark and parity indication); accordingly, those words longer than 6 characters have been abbreviated. Yet the extensive lines produced are wearisome to read, evincing a restricted lexicon and repetitive syntax.

However, it is at the level of coding that these lines reveal their formal innovations. The line generator was composed in a combination of FORTRAN-style pseudocode and assembly code; much of the programme is taken up with set-up information, instructions concerned with rendering the library data readable and with instructions concerning the format of the printed output, for example, "FORMAT(1H ,A6,1X,A6,3H / ,A6, 1X,A6,3H / ,A6)" produced the word order, forward-slashes and letter spacing. The necessity of such programming indicates the low level at which Coetzee was operating: he was having to write the subroutine to tell the computer to ignore blank lines in the input library, for example. Such programming might be "primitive" but much of it was written entirely from scratch in an extremely time-consuming exercise, heavily reliant upon a human coder. This programme does, however, seek to expand on the workings of its predecessor. Coetzee wrote two subroutines to automate aspects of the programme; the second was a random number generator (RGN). This programme utilised prime numbers to create the semblance of randomness: the numbers were used to automate word combination as Coetzee explored methods of writing that placed the author at one remove from the process of invention. In decoupling (albeit temporarily) creation from conscious authorial

[7] Backslashes are part of the computer-produced line.

intention, the programme offers a mechanical solution to what Eliot saw as the problem of personality in poetry.

Coetzee continued these experiments during his time at ICT, bolstered by the congenial atmosphere at Cambridge. There the Cambridge Language Research Unit, with which Coetzee's colleague Roger Needham was associated, was a pioneering centre of computational linguistics and Machine Translation. The unit was directed by Margaret Masterman, a philosopher taught by Wittgenstein and researcher in Natural Language Processing (NLP). Very much ahead of her time, Masterman used thesauri to design semantic-based models for machine translation – in direct opposition to Shannon's own contention that the "semantic aspects of communication are irrelevant to the engineering problem" (379). Masterman also wrote a programme to create computer-generated Haikus, work that Coetzee was certainly aware of by 1964 (he kept article clippings on the topic in a scrapbook). Meanwhile Philip Steadman, Stephen Bann and Mike Weaver were editing the first issue of FORM at Cambridge in 1965, which brought together articles on poetic imagery, "Computers and Design" (Computer Aided Design was just emerging at Cambridge in this period), modernist little magazines, architecture and French structuralism under the overall heading of "form", a topic that, as we have already seen, piqued Coetzee's interest thanks to his experiences in computational formalisations.

His experiments included attempts, written in Titan/Atlas 2 machine code, to produce RNG-composed lines of poetry. Using similarly repetitive, if slightly more complex grammatical structures, Coetzee deployed more exotic vocabularies to explore the degree to which semantic effect could be automated:

TURBID SCURF SADDEN THE CONVERSATION

WEEKLY ATARAXY TREPAN THE DOCUMENT

CARKING CREED VOICE THE ENTROPY

("Untitled (19 March 1965?) printout")

Or

IN THE BALLAD THE ANGEL NEEDS THE BIDDING . THE ADMONISHMENT IS SYMMETRICAL AND TASTE FILLS MY LAND

IN THE TWILIGHT THE POPPY CLAIMS THE CHILD . THE FLOWER IS HEAVY AND LONELINESS FILLS MY EARTH

IN THE WINTER THE FOOTFALL CHERISHS THE SNAIL . THE CICACA IS MILKY AND EFFENECENT FILLS MY FRUIT

("untitled printout dated 2 April 1965")

Or

THE DOCUMENT BEFORE THE HONEY TRAMPLES THE CAPTAIN OF CERTAINTY . THE SWEET WALKS

COUNTERFEIT DOCUMENT MAUDLIN HONEY CLOSE CAPTAIN INCONCLUSIVE SWEETS

THE LEAF OVER THE VIOLET WANTS THE NEST OF ANXIETY . THE BLOOM DRESSES

("Generate Index")

All three outputs were printed on different machines but do seem to draw upon the same base vocabulary, with its inclusion of a high percentage of unusual and arcane words. The number of lines generated was large; the second extract quoted above is taken from a text that extended across around 100 pages. Of these pages, the first few are annotated; a red or blue pen underlines various phrases and a tiny pencil tick mark accompanies them, suggesting that Coetzee was transcribing or otherwise recording for use elsewhere those phrases with which he was particularly taken: "the cicaca is

milky", "the chimera is sleeping" "the nude is punctual". As with his earlier experiments, the programmes here generate lines from which the poet might then identify choice phrases. The act of selection, such a key feature of the writing process for modernists like Pound and Ford, here attains full creative significance.

While these programmes might be classed as relatively unsophisticated (by Coetzee's proclaimed scale) they do demonstrate an increasing structural complexity – longer lines of output – and computational dexterity. The extant printouts indicate that Coetzee wrote some of this programming in machine code. While composing the RNG for the Titan/Atlas 2 Coetzee inserted an additional hand-written line in the Atlas assembly code print-out: "170 90 81 0 1C. if length>prime" ("Generate Index"). While the latter part of the line is human-readable commentary, the former is the machine-readable assembly code. At this point in his career Coetzee was working at such a low level of programming, writing the Supervisor code during the day, that, unsurprisingly, when he came to write computer poetry at night, he composed at the same deep level – utilising executable numerical notation. Despite its late-modernist associations, this was computer poetry composed with zero abstraction.

This was also dangerously executable code. The Titan/Atlas 2 had three separate protection regimes from which programmes could run: the regular user programme, the supervisor programme, and the interrupt code. While in the former the user could only modify memory storage areas assigned to them, the latter two provided enhanced access and execution possibilities (they could effectively do anything), but also brought with them a danger of a system crash, should a computer programme fail. Coetzee's programmes were written for the supervisor mode (indicated by using register "126" for the programme counter). Given his work on job control, Coetzee was naturally more familiar with this; however, his choice to run the

programme in that regime meant that no other client programmes could be run at the same time (no multiprogramming) and it also brought with it a small risk that the entire supercomputer could fail – at the height of the Cold War. But which computer did Coetzee use? By May 1965 the Titan and Atlas 2 teams were in the process of splitting, but the dates, and the paper records are inconclusive. While it is probable that these computer programmes were run on the Titan – Coetzee was unlikely to have had unfettered access to a supercomputer in an atomic weapons research facility at eleven at night (although this was the usual time to perform tests) – it is easy to imagine a different outcome.[8]

Remembering Computers in Texas

The creative and intellectual interests he developed during his time in the computing industry seem to have held Coetzee's attention across decades, continents and careers. However, it was at the University of Texas (UT) at Austin between 1965 and 1968 that Coetzee found a congenial intellectual context for pursuing research at the juncture of computational linguistics and literary study. The Linguistic Research Center (LRC) had opened at UT Austin under the directorship Winfred P. Lehmann in 1961. Like its Cambridge equivalent, it brought together linguists, philosophers, psychologists and computer scientists, and large amounts of defence funding, to develop machine-based translation and what one LRC staff member would label "text crunching" (Amsler). Lehmann (whose wife Rosamund lectured Coetzee in Old

[8] Moreover, Coetzee was rumoured to have been designated a "communist" by the South African government and deemed a security risk on the Aldermaston site. Landy, "Re: Coetzee...Again!" Regarding the incident in *Youth* where the narrator writes to Chinese Embassy offering his services (as an English teacher), I can find no evidence in the JMC archives to support or disprove its basis in Coetzee's own life, *Youth*, p. 153.

English[9]) offered an important role model: teaching across language and literature, he was a scholar of historical linguistics and machine translation who had put UT Austin on the map as a national centre for linguistics research and natural language processing (see Lehmann). Here Coetzee found a stimulating environment for pursuing those research interests developed while he was working and writing in the computer industry.

As a student Coetzee brought with him cutting-edge experience of working with supercomputer hardware and time-sharing multiprogramming. Although far from a backwater, in 1965 the university itself was not a major player in computer teaching and research. Coetzee's own knowledge marked him as an expert, explaining perhaps the lack of any computing courses on Coetzee's student record, despite his academic engagement with the subject at this time. While he was at UT Austin the university underwent a rapid expansion of its computer facilities, research programmes and course offerings. The Computation Center had been established in 1958 and a year after Coetzee arrived the powerful CDC 6600 supercomputer was installed. Significantly faster than the Atlas 2 and IBM stretch, although plagued by a poor operating system, the supercomputer enabled UT Austin to become a nationally recognised centre of computing. 1966 also saw the establishment of the department of Computer Science, which had a culture of disciplinary inclusivity and worked closely with the LRC. Early staff included NLP specialist Robert F. Simmons from RAND and James C. Browne, a frequent user of the Chiltern Atlas Laboratory in the UK. Teaching provision focused on graduate programmes. One of the first PhD students, Nell Boylan Dale would write a doctoral thesis that followed many of the same lines of enquiry Coetzee was considering: the contrast "quantitative" computational

[9] Kannemeyer conflates the couple in his account (147).

stylistics offered to the "impressionism" of traditional literary criticism (Dale 1, 2). The Computer Science department, along with the LRC and Computation Centre, while young would offer Coetzee an increasingly stimulating environment for conducting cross-disciplinary computational research during his doctoral studies.

Within the English Department itself, Coetzee pursued graduate courses with a heavy linguistic and stylistic bent. In addition to taking "Critical approaches to medieval literature" and "American fiction since James" he enrolled on "The study of style", "External and content structure in poetry, stylistics of prose" among others within the English Department and outside the department further courses on German and French, "Morphology and syntax", "Teaching English as a foreign language" and Dutch literature. Coetzee would read the work of Russian Formalists closely, demonstrating a commitment to pursuing methods of literary analysis that followed the "spirit of scientific positivism" (Eikenbaum, quoted in Gaskill 510). If his computer poetry experiments had highlighted that many of those features and practices in which modernism was deeply invested were constructed forms with specific real-world implications, Coetzee's engagement with style can be understood in part as an attempt to retain facets of this value system within a more quantitative context. These studies provided Coetzee with crucial grounding in the conceptual and methodological issues surrounding stylistic analysis, linguistics and literary representation across languages, offering a crucial complement to his already extensive knowledge of advanced programming. Together they would fuel fifteen years of research examining the application of computation to the study of literature and in particular style; as Johnston declares, "there is a strong... case for claiming J. M. Coetzee as a significant figure in the early development of digital humanities" ("Coetzee's Work in Stylostatistics").

The earliest academic example of this coalescence is a term paper entitled "Kinbote's Commentary on 'Pale Fire': An Essay in Stylistic Description and Interpretation". It is clearly influenced by Coetzee's reading in London and by the broader vogue in linguistics for generative grammar – Chomsky's computationalist *Aspects of the Theory of Syntax* was published in 1965 – and the high profile and well-funded efforts to mechanise translation, both of which were highlighting the rule-bounded nature of natural languages. If advanced computing could expose these underlying rules, might scholars of style benefit from these new methods? In his essay Coetzee seeks to explore precisely this topic, deploying statistical analysis of Nabokov's work to answer the question: "Is Kinbote's a new voice, or is he merely that other illustrious first-person narrator Humbert Humbert tricked out with a new tale"? ("Kinbote's Commentary" 2) Utilising mathematical formulae that flummoxed the essay marker, extensive diagramming of syntactical structures and pages of graphs, Coetzee offered extensive description of Charles Kinbote's prose style made on the basis of quantitative data. He relayed the count and standard deviation of syllables, clause, sentence and paragraph length, noun and verb ratios, diction probability and syntax structures, and statistical comparison with other datasets. In the process of concluding that "confession [Humbert] and commentary [Kinbote] are from the same hand" (11), Coetzee also made several methodological points: firstly that statistical analysis of literary texts usually involves a subjective selection of prose; and, secondly, that the most popular procedures utilised in such analysis are "extremely rudimentary, while more sophisticated ones entail massive amounts of computation" (3). In this early term paper, Coetzee was articulating both the promise and the limitations of his field of research.

[insert Fig 2.]

Coetzee's doctoral thesis, "The English Fiction of Samuel Beckett: An Essay in Stylistic Analysis", was the culmination of his interest in stylostatistics. The thesis sought to account for Beckett's increasing dissatisfaction with the English language (and his subsequent move to writing in French) via examination of the author's prose style. Later he would frame the linguistic side of this interest as the query:

> if a grammar of language is to be thought of as a formalisation of the processes required to decode utterances in that language, what *additional* capacities would such a grammar have to possess to allow us to decode utterances that employ novel or deviant rules? ("Linguistics and Literature" 42)

Applying stylostatistical methods to what the narrator of *Youth* would later describe as Samuel Beckett's "classless" prose, the thesis considered the insights that such "objective" criticism might offer to the "impressionistic" discipline of literary criticism (*Youth* 155). Like his term paper, Coetzee's thesis mapped syntactical units, syllabicity, sentence length, rarity of words and even Watt's chains of logical speculation, as Figure 2 demonstrates. Yet in this work and in published essays such as "Statistical Indices of 'Difficulty'" (1969) and "Samuel Beckett's 'Lessness': An Exercise in Decomposition" (1973) that derived from his studies, Coetzee ultimately concludes that the promise of objective critical insight is a hollow one.

For Coetzee, stylostatistics oversimplified thanks to being:

> dominated by a metaphor of linearity, a conception of language as a one-dimensional stream extending in time. The origin of the metaphor probably lies in our alphabet; it has been fortified by printing technology and by the twentieth-century metaphor of the mind as a computer with an input system which reads linear strips of coded information.

Evincing a keen awareness of the conceptual constraints such a metaphor places on the methodology, Coetzee makes an analogy with a material process with which he is very familiar to make two points. Firstly, in its computer input-like orientation towards the syntagmatic the metaphor ignores "precisely what we are interested in", namely "variation in a stylistic dimension". Secondly, this metaphor implies that the experience of literature is linear whereas Coetzee argues that it involves "incessant recursion": "as we read we are continually reformulating formal hypotheses to account for what we are reading and what we have read. Insofar as stylistic data have a formal function they too enter into these reformulations" ("The English Fiction of Samuel Beckett" 160-161). In rejecting the metaphor of linearity, Coetzee dismisses analogy with one computational process in favour of another: programming's recursive reading. He might criticise stylostatistics for its conceptual limitations, but Coetzee is very much open to the possibility that computation, and particularly programming, might offer positivist literary critical insight by formal analogy.

Formal, but not immaterial. Coetzee's understanding of the word, and by extension his understanding of the creative and critical processes that produce and interpret literature, is far from the dematerialised "Mathematical Theory of Communication", but rather accords with what we know about his own experiences with computers. I propose that Coetzee's programming work did much to underline

the *processes* of abstraction and formalisation that enabled computation and the underlying material structures on which they operated to function; these were processes and material structures that stylostatistics, dependent as it was on a statistical conception of information, deemed conceptually irrelevant (if not methodologically). It is thus notable that Coetzee's thesis turns on the juxtaposition of two definitions of style, at the heart of which is a question around (linguistic) media determinism. As Coetzee articulates it, to structural linguists such as Bernard Bloch "a word can be conveniently reduced, for the purposes of study, to a dimensionless and immaterial point. For Beckett, on the other hand, the 'terribly arbitrary materiality of the word's surface' is, we infer, at least in 1937, a burden" (2-3). While Beckett's prose style is mathematical and Watt analogous to Leibniz's automaton in being "something living encrusted on the mechanical", Coetzee is insistent that there is a "polar and antithetic relation" between the two notions of style (32, 2). In the terms of programming we could say that while Bloch happily approaches style through examination of the higher-level abstractions of FORTRAN, Beckett utterly rejects any correspondence between the machine code and its systematisation as a formal language; in other words, he dismisses the efficacy of the compiler, arguing for the imbrication of form and content.

Although well used to moving between levels of abstraction, Coetzee refuses to conceive of language, or the literary form and style it can produce, as dematerialised. It is no surprise that he turns to Beckett's manuscript revisions to *Watt*, the physical drafts of which were held in the UT Austin library, to think through the import of what he would describe here and in later articles as Beckett's career progression "toward a formalization or stylization of autodestruction" ("Beckett and the Temptations of Style" 45). In this he was placing himself in opposition not only to

Bloch, but to Shannon's mathematical theory of information, on which many of the early advances in computing derived. Far from conceiving of language as merely statistical, or as a transparent indicator of thought or perception – there are no "invisible extension cords leading from modes of thought or perception, into which a writer can plug his pen" ("The English Fiction of Samuel Beckett" 89) – Coetzee is interested in using computation to explore the specificity of authorial choices and word use in which material form is itself a consideration. More than this, what Coetzee would describe as Beckett's "automatism of style" would offer the young writer a foundation on which to build his own creative writing practice ("Beckett and the Temptations of Style" 49).

Coetzee and the Automatism of Writing

When he left Texas, Coetzee did not abandon his interests in the practical and speculative possibilities that computing might offer the critic and writer. Despite reservations about stylostatistics, his early years as an academic and published writer of fiction involved close contact with the field of humanities computing and the consequential development of his conception of aesthetic automatism.

In this period, Coetzee followed humanities computing research and attending seminars, in addition petitioning the MLA to upgrade the status of the seminar on Computer Applications and writing programmes to automate departmental administration at UCT where he taught from 1972 onwards.[10] In this new setting, Coetzee continued to perform computational analysis on texts. Filling notebooks with

[10] Examples include: "Final marks program," dated 19 Nov. 1984; "CT: Preregistration program," dated Sept. 1983?; "Univac 1100 at UCT," dated 4 Aug. 1983; "Computerizing Tutorial Allocations," dated 5 Mar. 1974; "Computer Team: Memorandum to all staff members involved in entering & checking marks," 18 Oct. 1976.

the "elementary" FORTRAN programming and punch card machine code necessary to examine Beckett's *Lessness*, Coetzee utilised UCT's Univac 1106 to execute the programme (Blue Notebooks). In 1973 his results were published, catching the attention of the *New York Times Review of Books* and *Scientific American* who announced "Beckett Safe From Computers" (Leonard 27). Later in the decade he also attended the Third International Conference on Computing in the Humanities to present a version of what would become his essay "Surreal Metaphors and Random Processes". This article draws explicitly on his own efforts at IBM and ICT to produce computer-generated metaphors. Comparing his programmes with the Surrealists' attempts to automate composition, he notes the "inevitable conflict between a systematizing theory of language [utilised in programming] and a literary practice (like that of Surrealism) with philosophical objections to the system". Quoting approvingly from George Steiner's *After Babel* that a "closed syntax, a formally exhaustible semantics, would be a closed world", Coetzee attempts to analyse the place of authorial consciousness and intention in creative writing (24). As he would later note about his own writing process, part of the activity for him is to overcome the "resistance" offered by the "automatism built into language: the tendency of words to call up other words, to fall into patterns that keep propagating themselves" (*Doubling* 18).

Coetzee would experiment with writing *with* automatism in his attempt at writing poetry based on computer line-generation. From the period between December 1971 and August 1975 Coetzee was working with lines generated on the Atlas 2, filling notebooks with selections from this material. Eventually published as "Hero and Bad Mother in Epic, a poem" in the black consciousness magazine *Staffrider* in 1978, for the best part of the decade beforehand Coetzee was studying,

selecting and re-ordering the computer-generated combinations, with an eye to the "inexhaustible" semantic effects of syntactical juxtapositions.[11] In this "epic" poem, as with his earlier work with surreal metaphors, Coetzee compares the procedures governing programming with those governing poetic writing. The published poem features alliteration, epic catalogues, epithets and kennings generated by the computer, offering what Zimbler has called a "phenomenology of metaphor" thanks to its unmasking of imaginative phrasing as rule-governed products (101). Yet the drafts, some of which were titled "beowulf", also experiment explicitly with representation forms more commonly used to display quantitative data (fig 3).

[insert fig. 3]

Here and in other drafts, Coetzee experiments with including parsing diagrams, tables, numbers ordered into inoperable algebraic formulae and the word "click" distributed over the page in a scatterplot ("beowulf"). Their inclusion suggests that Coetzee was exploring not only a "phenomenology of metaphor" but a phenomenology of representational systems; as Reuben Message has proposed, reading "Hero and Bad Mother" suggests that "what is disturbing about computer poetry is that it reveals that something of all writing is automatic, self-generative, and refuses comfortable reference" (Message 97). Certainly this accords with Coetzee's own description of writing quoted above. A deeply reflexive piece concerned with the function of poetic language and the procedures entailed in meaning making, "Hero

[11] For discussion of the imbrication of South African politics and computing in this period and its implications for literature, see my "Hero and Bad Motherland: J. M. Coetzee's Computational Critique".

and Bad Mother" offers the culmination of Coetzee's engagement with computer poetry and automatized writing.

While it would be almost twenty-five years, in which he would author more than a dozen books, before Coetzee would publish another creative work explicitly engaged with computing, his conception of literary language and his own literary style were likely influenced by these early experiments with computing. His style might not be as permutational as that of Beckett's late English prose, but for Coetzee creative writing is often defined in productive resistance to writing in which the "machine runs the operator" ("Note on Writing" 95).

Works Cited

Archival Material:

J. M. Coetzee Papers. Harry Ransom Center, University of Texas at Austin. Accessed Jan. 2015 and Nov. 2016.

Annotations to "Christopher Alexander *Notes on the Synthesis of Form,*" Research materials: reading notes for *Dusklands,* 1960s. Container 99.3.

"beowulf", draft of "Hero and Bad Mother", dated 24 Oct. 1972. "Hero and Bad Mother" (poem), typed and handwritten drafts and correspondence, 1972-1975. Container 113.11.

Blue Notebooks. Typescripts: early works, 1972-73. Container 63.5.

"Computer Team: Memorandum to all staff members involved in entering & checking marks" 18 Oct. 1976. English tutorials, typed and handwritten documents, 1978-1982. Container 113.5.

"Computerizing Tutorial Allocations," 5 Mar. 1974. English tutorials, typed and handwritten documents, 1978-1982. Container 113.5.

"CT: Preregistration program," Sept. 1983?. English tutorials, typed and handwritten documents, 1978-1982. Container 113.5.

"Dissertation Abstracts." Notes for Master's Research, circa 1962. Container 121.1.

"Final marks program," 19 Nov. 1984. English tutorials, typed and handwritten documents, 1978-1982. Container 113.5.

"Ford Madox Ford, *Henry James: A Critical Study* (London, 1913, 192pp)." Notes for Master's Research, circa 1962. Container 121.1.

"Generate Index", 25 Mar. (or May?) 1965. Part of file labelled "Computer Poetry, Dot Matrix printout. 1965". Container Obs 143.

"Kinbote's commentary on 'Pale Fire': An essay in stylistic description and interpretation". Term Papers for Graduate Courses. Container 117.7.

Letter from J. M. Coetzee to Stanley Skewes, 6 Dec. 1964. Applications to graduate school, correspondence to and from various universities, 1964-1965. Container 104.1.

"Line Generator," 30 May 1963, Part of file labelled "Computer Poetry, Dot Matrix printout. 1965". Container Obs 143.

"Univac 1100 at UCT", 4 Aug 1983. English tutorials, typed and handwritten documents, 1978-1982. Container 113.5.

Untitled and undated (19 Mar. 1965?) printout. Part of file labelled "Computer Poetry, Dot Matrix printout. 1965". Container Obs 143.

Untitled printout, 2 April 1965. Part of file labelled "Computer Poetry, Dot Matrix printout. 1965". Container Obs 143.

Secondary Sources:

Amsler, Robert A. "Re: The History of the LRC and Computing at UT." Received by Rebecca Roach, 12 Nov. 2016.

Attridge, David. *J. M. Coetzee and the Ethics of Reading: Literature in the Event*. U of Chicago P, 2004.

Attwell, David. *J. M. Coetzee & the Life of Writing: Face to Face with Time*. Oxford UP, 2015.

Burke, Damien. "Tsr-2". Personal message received by Rebecca Roach, 12 Oct. 2017.

Ceruzzi, Paul E. *A History of Modern Computing*. MIT P, 1998.

Coetzee, J. M. "Achterberg's 'Ballade van de gasfitter': The Mystery of I and You." *PMLA.* vol. 92, no. 2, 1977, pp. 285-296

---. "A Note on Writing." Reprinted in *Doubling the Point*, pp. 94-95.

---. "Beckett and the Temptations of Style." 1973. Reprinted in *Doubling the Point,* pp. 43-49.

---. "Computer Poem." *The Lion and the Impala.* vol. 2, no. 1, 1963, pp. 12-13.

---. "The English Fiction of Samuel Beckett: An Essay in Stylistic Analysis." Dissertation, U of Texas at Austin, 1969.

---. "Linguistics and Literature." *An Introduction to Contemporary Literary Theory*, edited by Rory Ryan and Susan Van Zyl. A. D. Donker, 1982, pp. 41-52.

---. "The First Sentence of Yvonne Burgess' *The Strike.*" 1976. Reprinted in *Doubling the Point*: 91-93.

---. *Doubling the Point: Essays and Interviews*. Edited by David Atwell. Harvard UP, 1992.

---. *Dusklands*. 1974. Vintage, 1998.

---. *In the Heart of the Country*. 1976. Vintage, 2004. And ,

---. Review of *Strange Attractors: Poems of Love and Mathematics*, edited by Sarah Glaz, Joanne Growney. *Notices of the American Mathematics Society,* vol. 56, no. 8, 2009, pp. 944-946.

---. "Surreal Metaphors and Random Processes." *Journal of Literary Semantics*, vol. 8, no. 1, 1979, pp. 22-30.

---. "The Work of Ford Madox Ford with Particular Reference to the Novels." MA thesis, U of Cape Town, 1963.

---. *Youth*. 2002. Vintage, 2003.

Dale, Nell Boylan. “Computational stylistics: A Methodological Study.” Dissertation, U of Texas at Austin, 1972.

Edwards, Paul N. *The Closed World: Computers and the Politics of Discourse in Cold War America*. MIT P, 1996.

Emerson, Lori. “Materiality, Intentionality, and the Computer-Generated Poem: Reading Walter Benn Michaels with Erin Mouré’s *Pillage Laud.” English Studies in Canada*, vol. 34, no. 4, 2008, pp. 45-69.

Ensmenger, Nathan. *The Computer Boys Take Over: Computers, Programmers, and the Politics of Technical Expertise*. MIT P, 2010.

Gaskill, Nicholas. “The Close and the Concrete: Aesthetic Formalism in Context.” *New Literary History*, vol. 47, no. 4, 2016, pp. 505-524.

Gitelman, Lisa. *Paper Knowledge: Toward a Media History of Documents*. Duke UP, 2014.

---. “‘Materiality has always been in play’: an interview with N. Katherine Hayles. *Iowa Journal of Cultural Studies*, vol. 2, 2002, pp. 7-12.

James, David. *Modernist Futures: Innovation and Inheritance in the Contemporary Novel*. Cambridge UP, 2012.

Johnston, Peter. “J . M. Coetzee’s Work in Stylostatistics.” *Digital Humanities Quarterly*, vol. 8, no. 3, 2014, http://www.digitalhumanities.org/dhq/vol/8/3/000188/000188.html. Accessed 22 June 2015.

---. “‘Presences of the Infinite’: J. M. Coetzee and Mathematics.” Dissertation, Royal Holloway, University of London, 2013.

Kannemeyer, J. C. *J. M. Coetzee: A Life in Writing.* Translated by Michiel Heyns. Scribe, 2012.

Kenner, Hugh. “Art in a Closed Field.” *The Virginia Quarterly Review*, vol. 38, no. 4, 1962, pp. 597-613.

Kevin Maney, Steve Hamm, and Jeffrey M. O'Brien. *Making the World Work Better: The Ideas that Shaped a Century and a Company*. IBM P, 2011.

Kirschenbaum, Matthew and Sarah Werner. “Digital Scholarship and Digital Studies: the State of the Discipline” *Book History*, vol. 17, 2014, pp. 406-458.

Knuth, Donald and Luis Trabb Pardo. “The Early Development of Programming Languages.” Computer Science Department, Stanford University, August 1976. ftp://bitsavers.informatik.uni-stuttgart.de/pdf/stanford/cs_techReports/STAN-CS-76-562_EarlyDevelPgmgLang_Aug76.pdf. Accessed 10 Feb. 2017.

Landy, Barry “Atlas 2 at Cambridge Mathematical Laboratory (And Aldermaston and CAD Centre).” Atlas Symposium, School of Computer Science, University of Manchester, 5 Dec. 2012. PDFs of the talks are available at http://elearn.cs.man.ac.uk/~atlas/. Accessed 8 Jan. 2015.

---. “Re: Coetzee... Again!” Received by Rebecca Roach, 1 May 2017.

Lavington, Simon. “Ferranti Atlas 2 computers at AWRE Aldermaston and at the CAD Centre.” Atlas Symposium.

---. “The Atlas story.” Atlas Symposium, second edition, 6 Dec. 2012.

Lehmann, Winfred P. “My Early Years in Machine Translation.” *Early Years in Machine Translation: Memoirs and Biographies of Pioneers,* edited by W. John Hutchins. John Benjamins, 2000, pp. 147-164.

Leonard, John. “Last Word: Beckett Safe from Computers.”, *New York Times Review of Books*, 19 Aug. 1973, p. 27.

Levine, Caroline. *Forms: Whole, Rhythm, Hierarchy, Network* Princeton: Princeton UP, 2004.

Liu, Lydia H. *The Freudian Robot: Digital Media and the Future of the Unconscious*. U of Chicago P, 2010.

Love, Heather A. "Cybernetic Modernism and the Feedback Loop: Ezra Pound's Poetics of Transmission." *Modernism/modernity* vol. 23, 2016, pp. 89–112.

Manovich, Lev. "Aesthetics, 'Formalism,' and Media Studies." *Keywords in Media Studies,* edited by Laurie Oellette and Jonathan Gray. New York UP, forthcoming 2017.

McHale, Brian. *Constructing Postmodernism.* Routledge, 1992.

Message, Reuben. "Reading, Deconstruction and Technological excess: some thoughts on J. M. Coetzee's 'Hero and bad Mother in Epic, a poem'." *Synthesis, Journal of Humanities and Social Affairs,* vol. 2, no. 1, 2010, pp. 83-101.

Needham, Roger. Letter to the Editor, *Times Literary Supplement*, 8 June 1962, p. 429.

Pressman, Jessica. *Digital Modernism: Making It New in New Media*. Oxford UP, 2014.

Pyle, Ian. "Appendix: About the Atlas supervisor" in "Memories of Atlas Fortran." Symposium.

Reference Manual: 709/7090 FORTRAN Programming S*ystem*, IBM, 1961.

Roach, Rebecca. "Hero and Bad Motherland: J. M. Coetzee's Computational Critique" Forthcoming.

Shannon, Claude E. "A Mathematical Theory of Communication." *The Bell System Technical Journal*, vol. 27, no. 3, 1948, pp. 379-423.

Zimbler, Jarad. *J. M. Coetzee and the Politics of Style*. Cambridge UP, 2014.

Funding Note:

Working across disciplinary boundaries is always a productive challenge and leads to many debts of gratitude. I would like to thank David Illsley, Chris Kerr, Simon Lavington, Victoria Marshall, Martin Roach and Tom Roach for their correspondence and conversations around computer programming (wrong end of the stick moments are entirely attributable to me) and to Barry Landy who patiently explained Atlas Autocode to me. This research benefitted enormously from a fellowship at the Harry Ransom Center, University of Texas at Austin in 2016, where Abby Adams and Rick Watson were extremely helpful in making Coetzee's digital archives available; so too my thanks go to Jim Browne and Nell Dale who agreed to talk to me about their time at the Department of Computer science and to Jessica Meyerson at the Briscoe Center for American History, and the network of respondents who provided me with a wealth of information about early computing at UT Austin. This research **wa**s funded by a European Research Council Grant, 'Ego-media: the impact of new media on forms and practices of self-presentation' [no. 340331]**.**